

REMARKS

Applicants have reviewed and considered the Office Action dated April 28, 2007 and the cited references therein. In that Office Action, the Examiner rejected Claims 1, 3-8, 10-53, 55, 57-59, and 61-70 under 35 U.S.C. § 103(a). In view of the following remarks, Applicants respectfully request reconsideration and allowance of the pending claims.

Applicants respectfully believe that the Examiner has misunderstood at least a portion of Applicants' remarks in the response to the previous Office Action dated September 28, 2006. Therefore, Applicants address the remarks and each of the Examiner's responses in detail below.

Rejection under 35 U.S.C. § 103

Claims 1, 3-8, 10-53, 55, 57-59 and 61-70 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Filepp et al. (US 5,347,632) in view of Campbell et al. (US 6,920,615). Applicants respectfully traverse the rejection for at least the following reasons.

Filepp and Campbell are Not Combinable

The Examiner has responded to Applicants' remarks that Filepp and Campbell are not combinable by stating:

While the applicant feels that the present invention is directed toward a thin client architecture and that Filepp teaches away from this, it is noted that a thin client architecture is not a limitation of the claims. The claims simply state an application that is "server-based" and "client-side controlled" and it is maintained that Filepp's programs and applications meet these limitations. *OA dated Apr. 28, 2007, p. 20.*

Fat Client Architecture v. Thin Client Architecture

Applicants respectfully aver that the phrases "fat client architecture" and "thin client architecture" were used by Applicants in the previous response merely for describing each system (i.e., Filepp, Campbell, and Applicants' system) in order to better aid the Examiner in understanding Applicants' remarks. These phrases have become known as terms of art to represent two types of system architectures. Fat client architecture includes an architecture

wherein substantial proportions of the processing are performed on the client-side to reduce the load on the server. Thin client architecture includes an architecture wherein substantial proportions of the processing are performed on the server-side to reduce the load on the client, e.g., executing a server-based, client-controlled application at a user interface UI server. Thus, Applicants believed use of the terms “fat client architecture” and “thin client architecture” in the previous response would be easier to read and keep track of than repeated use of the phrases “client-based to reduce the load on the server” and “server-based to reduce the load on the client,” respectively.

As previously stated, the present invention is directed towards a thin client architecture. Although the Examiner asserts that a thin client architecture is not a limitation of the claims, Applicants respectfully assert that “server-based” and “client-side controlled” as provided in the claims are included in the term “thin client architecture,” as it is used in the art. In contrast, a fat client architecture includes a client-based and client-side controlled application.

Filepp is directed towards an architecture that reduces the load on the server and provides for a fat client:

. . . the invention includes procedures for formulating objects that have been specially structured to include display data, control data and program instructions for supporting the applications at the network reception systems, the objects being pre-created, parceled units of information that may be distributed and stored at lower levels in the network . . . so as to reduce processing demand on the network higher element . . . *Filepp, Col. 2, l. 61-Col. 3, l. 1* (emphasis added).

The object of reception system software is to minimize mainframe processing . . . *Filepp, Col. 82, ll. 18-19* (emphasis added).

More particularly, Filepp describes a method of distributing application partitions to client devices for execution of the application at the client devices:

Objects provide a means of packaging and distributing partitioned applications . . . objects make up one or more partitioned applications, and are retrieved on demand by a user's RS 400 for interpretive execution and selective storage. *Filepp, Col. 5, ll. 40-44* (emphasis added).

Each application partition is an independent, self-contained unit and can operate correctly by itself. *Filepp*, Col. 5, ll. 27-28.

[I]t is desirable to have as many of these program objects staged for execution at or as close to RS 400 as possible. *Filepp*, Col. 11, ll. 61-63; see *Filepp*, Col. 11, l. 42 (“Program objects 508 contain program instructions . . .”).

That is, the Filepp system delivers unexecuted program applications, in partitions, to the client device for assembly of the partitions and execution of the program applications at the client device. Even though the programs may be stored at the server, they are executed at the client device. Thus, the processing load on the server is reduced.

Furthermore, “[I]f proposed modification would render the prior art invention being modified unsatisfactory for its intended purpose, then there is no suggestion or motivation to make the proposed modification.” *MPEP 2143.01(V)* (citing *In re Gordon*, 733 F.2d 900 (Fed. Cir. 1984)). By combining Filepp, a fat client architecture, with Campbell, a thin client architecture, the Examiner is eliminating the intended purpose and benefits/advantages of the fat client disclosed in Filepp. That is, modifying the fat client disclosed in Filepp with the teachings disclosed in Campbell will increase the load on the server.

Accordingly, whereas Filepp is directed towards an architecture that reduces the load on the server and provides for a fat client (i.e., client that performs the bulk of the data processing operations), the present invention is directed towards an architecture that reduces the load on the client and provides for a thin client (i.e., server that performs the bulk of the data processing operations). See *Specification*, p. 2, para. [0021]; p. 8, para. [0086].

Applicants still maintain that Filepp is not combinable with Campbell or any other reference to obviate Applicants’ claimed invention. However, even if the references are combinable, Applicants assert that neither Filepp nor Campbell, alone or in combination, disclose, teach, or suggest the Applicants’ claimed invention.

Formatting Characteristics Based Upon Device Capabilities

The Examiner has further mistakenly separated one of Applicants’ arguments into two separate arguments (i.e., Examiner’s labeled Arguments 2 and 4). As such, the Examiner stated

“[t]he applicant is reminded that one cannot show nonobviousness by attacking references individually where the rejections are based on a combination of references.” *OA dated Apr. 28, 2007, p. 21*. Applicants respectfully point out that these arguments address the same patentably distinguishable limitation of the claims, one argument with respect to Filepp and the other with respect to Campbell. Applicants believe that some confusion may have arisen in the order the remarks were presented in Applicants’ previous response. Applicants did not attack each reference separately but completely discussed Filepp before discussing Campbell. Applicants have herein reordered the remarks, clarified the patentably distinguishable limitations, and responded to the Examiner’s response in the outstanding Office Action in order to aid the Examiner’s understanding of the novelty and nonobviousness of Applicants’ claimed invention.

Filepp

Applicants previously asserted and continue to assert that Filepp fails to describe every element of every claim “in as complete detail” as is contained in the claims, as required by MPEP § 2131. Specifically, independent Claim 1 recites “formatting characteristics of said intermediate UI based upon a number of device capabilities for said client device.” Independent Claim 19 recites “defining a user interface (UI) form in response to a number of device capabilities for a client device.” Independent Claims 38, 45, and 53 provide similar recitations. *See* Claim 38 (“according to a UI format determined by a UI server and based upon a number device capabilities for said client device”); Claim 45 (“generating a user interface (UI) form definition for a server-based application based upon a number of device capabilities for a client device”); Claim 53 (“a UI formatting module that generates said UI form definition based upon a number of device capabilities for a client device”). Not only does Filepp fail to teach this limitation, but the Examiner has asserted the same view:

Filepp did not explicitly state formatting characteristics of said intermediate UI based upon a number of device capabilities for said client device. *OA dated September 28, 2006, page 3.*

Additionally, a review of Filepp reaffirms the position that the reference fails to disclose, teach, or suggest the limitation of defining or generating a user interface form based upon or in response to a number of device capabilities for a client device. Filepp discloses the same

information and structure for “objects” regardless of the client device capabilities. *See Filepp, Col. 7, ll. 24-25* (“There are two types of information in the network which are utilized by the RS 400: objects and messages.”). Filepp discloses that objects contain “information about what is to be displayed and how it is to be displayed,” but nowhere does Filepp disclose, teach, or suggest that these objects are “based upon” or “in response to” client device capabilities or provided by the client device OS or applications as claimed. *See Filepp, Col. 7, ll. 24-46; Col. 7, l. 64-Col. 8, l. 39* (describing the objects of Filepp without any indication of correspondence to client device capabilities).

In fact, Filepp describes a client device that provides a:

. . . common interface to other elements of interactive computer network 10 . . . and a common protocol for user application conversation which is independent of the personal computer brand name used. RS 400 thus constitutes a universal terminal for which only one version of all applications on network 10 need be prepared . . . Objects have a uniform, self-defining format . . . *See Filepp, Col. 4, l. 61-Col. 5, l. 7.*

The Examiner asserts that although Applicants argue that Filepp’s objects have a uniform, self-defining format, not a format depending upon a particular client device, there is no explicit teaching in Filepp that states that these forms cannot be manipulated. *OA dated Apr. 28, 2007, p. 20.* Applicants respectfully assert that the opposite is also true – that there is no explicit teaching in Filepp that states these forms can be manipulated. The Examiner is merely using impermissible hindsight to modify the system of Filepp to obtain each of the limitations of Applicants’ claimed invention.

Furthermore, the Examiner asserts that there is no explicit teaching in Filepp that one of ordinary skill in the art would not find an advantage to such manipulation (i.e., using the teachings of Campbell). *OA dated Apr. 28, 2007, p. 20.* However, Applicants respectfully refer the Examiner to the above remarks provided by Applicants relating to fat client architecture. Filepp is directed to a fat client architecture. A fat client architecture reduces the load on the server. Filepp discloses a system that does this by providing common applications, in partitions, to the client devices, where the partitions are combined and the applications are executed. Filepp cannot reduce the load on the server if Filepp has to rewrite every application based on the

capabilities of each client device. This is a large reason Applicants insist that Filepp is not combinable with Campbell or any other reference relating to thin client architecture. Filepp can simply not be modified with the teachings from Campbell, or anywhere else, in such a manner that would increase the load on the server. Such a combination defeats the intended purpose of the Filepp system.

Accordingly, Filepp not only fails to disclose, teach, or suggest formatting, defining, or generating a user interface form based upon or in response to a number of device capabilities for a client device, but, to the contrary, each client device provides a “common interface” such that other elements of the network would have no need to generate user interface forms based on particular client capabilities. Indeed, the objects have a “uniform, self-defining format,” not a format dependant upon a particular client device. The disclosure of Filepp, therefore, teaches away from the claimed limitation.

Campbell

Besides not being combinable with Filepp, Campbell fails to remedy the deficiencies of Filepp. Campbell discloses a method and system for dynamic services support, including a portal-page service and interface bundle installed at a gateway. *Campbell, Col. 1, ll. 43-48*. “The portal-page service searches for the customer service and generates a user interface for the customer service based on the customer service.” *Id.* Campbell further discloses that the portal service “may present an appropriate web page based on the type and capabilities of . . . [the] user device.” *Campbell, Col. 14, ll. 49-51*. Similarly, Campbell discloses that “[p]ortal service may . . . present an interface appropriate for the small screens typically associated with PDAs.” *Campbell, Col. 14, ll. 55-57*.

Nonetheless, Campbell does not disclose, teach, or suggest formatting, defining, or generating a user interface based upon or in response to a number of device capabilities for a client device, where the user interface, as recited in Applicants’ independent claims, contains a user interface form or skeletal user interface and controls, icons, labels, or menu items, each being independently updateable. It may be one concept, as disclosed in Campbell, to change a user interface for appropriate display on a user device, but it is an entirely novel and inventive

concept to generate a user interface, based upon device capabilities for a client device, from independently updateable forms, controls, icons, labels, or menu items, which Campbell does not disclose. Applicants respectfully assert that the Examiner is merely using impermissible hindsight to combine Campbell with Filepp in order to obtain each of the limitations in Applicants' claims.

Maintaining a Shadow Cache

Applicants also previously argued that neither Filepp nor Campbell, alone or in combination, disclose maintaining a shadow cache. Applicants respectfully believe that the Examiner's response in the outstanding Office Action mistakenly confuses a shadow cache with standard cache memory.

Filepp

Filepp fails to disclose, teach, or suggest maintaining a shadow cache at the UI server, said shadow cache including a list of source data items transmitted from said UI server to a client device as recited in independent Claims 1, 19, 38, 45, 53, and 59. *See* Claim 1 ("maintaining a shadow cache at said UI server, said shadow cache including a list of source data items transmitted from said UI server to said client device"); Claim 19 ("maintaining a shadow cache at said UI server, said shadow cache including a list of source data items transmitted from said UI server to said client device"); Claim 38 ("maintaining a shadow cache at said UI server, said shadow cache including a list of source data items transmitted from said UI server to said client device"); Claim 45 ("maintaining a shadow cache at said UI server, said shadow cache including a list of source data items transmitted from said UI server to said client device"); Claim 53 ("a shadow cache module, including a list of source data items transmitted from said UI server to said client device"); Claim 59 ("maintain a shadow cache, said shadow cache including a list of source data items transmitted from said UI server to said client device").

Applicants' disclosure recites:

The UI server application includes or is otherwise associated with ... a shadow cache element. *Specification, p. 7, para. [0081]*.

Shadow cache, which is maintained by the UI server, may include a list of source data items, UI form information, and other client-related data that has been transmitted from the UI server to the client device . . . the shadow cache may contain data representing items transmitted from the UI server to the client device and/or items that have been saved in the client cache . . . The UI server can interrogate the shadow cache to determine the data cached at the client device, and update the shadow cache in response to modifications to cached data entered by the client device. Shadow cache allows the UI server to monitor the status of the client cache, maintain synchronization with the client device, recognize when it is appropriate to “push” certain data types to the client device, support the persistent application states, and allows the UI server application to manage the downloading of new or modified information to the client device without repeatedly invoking applications. *Specification, p. 9-10, para. [0096].*

The Examiner repeatedly points to Filepp, Col. 7, ll. 17-41 and Col. 5, ll. 20-25 for support of the assertion that Filepp discloses this limitation. Specifically, the Examiner states “Filepp clearly utilizes a plurality of cache/concentrator units.” *OA dated Apr. 28, 2007, p. 21.* Although Filepp discloses the use of cache/concentrator units, there is no disclosure of any use of the cache/concentrator units for performance of any task other than aiding the delivery of requested data to the RS 400s. *Filepp, Col. 7, ll. 13-16.* Specifically, the information used in a RS 400 either resides locally at the RS 400 or is available on demand from the cache/concentrator units or file server. *Filepp, Col. 7, ll. 18-20.* Filepp, however, does not disclose use of the cache/concentrator units in any other manner. Particularly, Filepp does not disclose a shadow cache as disclosed in Applicants’ specification or claims. That is, Filepp does not disclose a shadow cache, including a list of source data items transmitted from said UI server to a client device.

Therefore, it remains unclear to Applicants what the Examiner refers to in these sections for support since these sections are merely directed to the disclosure of standard cache memory. In fact, nowhere in Filepp does it disclose, teach, or suggest maintaining a shadow cache at the UI server as disclosed and taught in the present application.

Campbell

Campbell fails to remedy the deficiencies of Filepp. Campbell fails to disclose, teach, or suggest maintaining a shadow cache at the UI server, said shadow cache including a list of source data items transmitted from said UI server to a client device. As stated above, Applicants disclose a shadow cache, which is maintained by the UI server. The shadow cache “may include a list of source data items” and “may contain data representing items transmitted from the UI server to the client device and/or items that have been saved in the client cache . . .” *Specification, p. 9-10, para. [0096]*. Campbell does not disclose a shadow cache, or anything similar to the shadow cache, as it is disclosed in Applicants’ specification and claims.

Summary

Therefore, neither Filepp nor Campbell, alone or in combination, disclose, teach, or suggest the invention of independent Claims 1, 19, 38, 45, 53, and 59. As discussed above, neither Filepp nor Campbell disclose, teach, or suggest defining or generating a user interface form based upon or in response to a number of device capabilities for a client device. Similarly, neither Filepp nor Campbell disclose, teach, or suggest maintaining a shadow cache at the UI server. The remaining pending claims depend from the patentable independent claims and are therefore also patentable over Filepp and Campbell. Accordingly, Applicants respectfully request reconsideration and withdrawal of the obviousness rejection of pending Claims 1, 3-8, 10-53, 55, 57-59, and 61-70.

Conclusion

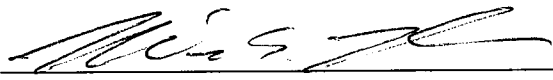
This application now stands in allowable form and reconsideration and allowance is respectfully requested.

This response is being submitted on or before August 28, 2007 with a three month request for an extension of time and the required fee of \$1,020.00. It is believed that no additional fees are due in connection with this filing. However, the Commissioner is authorized to charge any additional fees, including extension fees or other relief which may be required, or credit any overpayment and notify us of same, to Deposit Account No. 04-1420.

Respectfully submitted,

DORSEY & WHITNEY LLP
Customer Number 25763

Date: August 2, 2007

By: 
Min (Amy) S. Xu, Reg. No. 39,536
(612) 752-7367